

## Rancang Bangun *Data Warehouse* Perpustakaan untuk Perguruan Tinggi

Moch.Ali Machmudi<sup>1</sup>

<sup>1</sup>Prodi Manajemen Informatika, STMIK Bina Patria Magelang  
Jl. Raden Saleh No.02 Magelang  
E-mail: aliadhinata@gmail.com<sup>1</sup>

### Abstract

The usage of Computer Based Information System in the library which has been in progress for long time will cause the addition of big unuseful transaction data. The addition of data will be useful if we build warehouse data. This research is developed a design of warehouse data of library in educational institution. The process of development will be done by prototyping model. By using features of DBMS, the process of extraction, transformation and loading can done effectively and efficiently. Besides that, the process of information delivery can be done via available features. The availability of warehouse data will enable to be done various analysis which will generate useful information. Information will be easier to be read and understood will be performed in the form of graphic. Further, because of available warehouse data, can be developed data mining application.

**Key Words:** data warehouse, DBMS, prototyping

### Abstrak

Penggunaan Sistem Informasi Berbasis Komputer (Computer Based Information System/ CBIS) pada sebuah perpustakaan yang sudah berlangsung dalam kurun waktu yang lama akan menyebabkan terjadi penumpukan data transaksi yang sudah tidak terpakai dalam jumlah yang besar. Penumpukan data ini akan dapat akan bisa bermanfaat jika dikelola, misalnya dengan membangun sebuah data warehouse. Dalam penelitian ini dikembangkan sebuah rancang bangun data warehouse perpustakaan yang dapat dipergunakan untuk pengembangan aplikasi data warehouse untuk perpustakaan di institusi pendidikan. Proses pengembangan dilakukan dengan menggunakan model prototyping. Dengan memanfaatkan fitur-fitur yang dimiliki oleh DBMS, proses ekstraksi, transformasi serta loading atau pengisian data dapat dilakukan secara efektif dan efisien. Selain itu, proses information delivery pun dapat dilakukan dengan menggunakan fitur yang tersedia. Adanya data warehouse akan memungkinkan untuk dilakukan berbagai analisa yang akan menghasilkan informasi yang berguna. Informasi akan lebih mudah dibaca dan dipahami maka dapat ditampilkan dalam bentuk grafik. Lebih lanjut, dengan adanya data warehouse, maka dapat dikembangkan aplikasi data mining.

**Kata Kunci:** data warehouse, DBMS, prototyping

### 1. PENDAHULUAN

Pada beberapa dekade terakhir, penggunaan sistem informasi berbasis komputer (*Computer Based Information System/ CBIS*) merupakan hal yang sudah umum dilakukan. Penggunaan

CBIS dimaksudkan untuk efektivitas dan efisiensi organisasi dalam melakukan kegiatan operasionalnya, yang pada akhirnya dapat meningkatkan produktivitas dan menjaga kelangsungan

hidup organisasi. Penggunaan CBIS dalam kurun waktu yang lama berdampak pada diperlukannya mekanisme bagi organisasi dalam mengelola data-data yang sudah tidak terpakai dalam kegiatan operasional sehari-hari. Pada umumnya organisasi melakukan proses *back up* data lama ke dalam suatu penyimpanan tertentu.

Dengan adanya penyimpanan data dalam jumlah yang besar ini kemudian memunculkan ide atau gagasan untuk menggali berbagai informasi yang terkandung dalam data-data tersebut. Data-data yang disimpan dalam kurun waktu tertentu dan dalam jumlah yang besar, memungkinkan untuk dianalisa dan digali berbagai informasi yang terkandung di dalamnya. Penggalan informasi ini tentu saja akan mendatangkan berbagai manfaat yang dapat berguna bagi organisasi.

Perpustakaan merupakan salah satu komponen yang wajib ada dalam sebuah organisasi pendidikan. Perpustakaan secara tradisional, selain merupakan tempat di mana disimpan koleksi berbagai buku, majalah, artikel dan lain sebagainya, juga merupakan pusat kegiatan bagi siapa saja yang ingin membaca, mencari literatur, maupun kegiatan lain terkait dengan studi pustaka. Dengan pencatatan transaksi pada perpustakaan berbasis CBIS dalam kurun waktu yang lama, maka dapat dipastikan bahwa akan terdapat data yang disimpan dalam jumlah yang sangat besar. Jika data ini dikelola menjadi sebuah gudang data atau *data warehouse* tentu saja akan memungkinkan bagi pihak pengelola perpustakaan untuk menggali berbagai informasi yang terkandung di dalamnya, yang pada akhirnya dapat mendatangkan berbagai nilai manfaat bagi kemajuan perpustakaan tersebut. *Data warehouse* menurut Poe (1998) merupakan metode dalam perancangan *database*, yang menunjang DSS (*Decision Support System*) dan EIS (*Executive Information*

*System*). Secara fisik data warehouse adalah database, tapi perancangan data warehouse dan database sangat berbeda. Dalam perancangan database tradisional menggunakan normalisasi.

## 2. METODE PENELITIAN

Metode yang dipergunakan dalam proses merancang bangun perpustakaan adalah menggunakan model *prototyping*. Model *prototyping* dimulai dengan pengumpulan persyaratan. pengembang dan pelanggan bertemu dan menentukan tujuan umum untuk perangkat lunak, mengidentifikasi persyaratan apapun yang dibutuhkan. Desain cepat berfokus pada representasi aspek-aspek dari perangkat lunak yang akan terlihat oleh pelanggan / pengguna misalnya, pendekatan input dan format output (Pressman, 2001).

### 2.1. Identifikasi Kebutuhan

Analisa kebutuhan digunakan untuk mengetahui dan menterjemahkan semua permasalahan serta kebutuhan perangkat lunak dan kebutuhan system yang dibangun. Oleh karena itu, dalam tahap ini dilakukan proses pengumpulan data untuk membangun sistem. kebutuhan-kebutuhan yang diharapkan dengan adanya *data warehouse* perpustakaan adalah sebagai berikut:

- 1) Dibutuhkannya aplikasi yang akan digunakan sebagai pangkalan data-data yang berisi mengenai informasi perpustakaan yang dapat digunakan sebagai dasar pengambilan keputusan strategis berikutnya.
- 2) Dibutuhkan sebuah fasilitas untuk menghasilkan informasi yang merupakan rangkuman yang ditampilkan dalam bentuk tabel maupun tampilan visual dalam bentuk grafis. Informasi yang ditampilkan berupa rangkuman transaksi peminjaman yang dilihat dari berbagai aspek seperti asal fakultas peminjam, asal program studi peminjam, judul buku,

kategori buku serta profile peminjam. Rangkuman yang dibuat haruslah mempertimbangkan dimensi waktu.

Berdasarkan uraian di atas, maka dapat dibuat daftar kebutuhan sistem seperti disajikan dalam tabel 1.

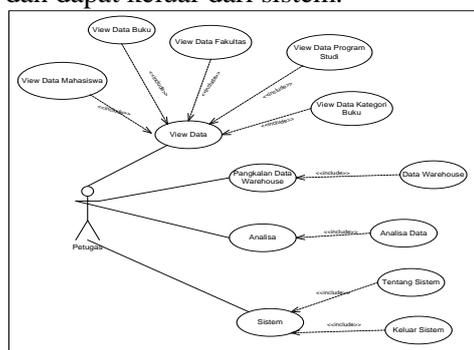
Tabel 1 Daftar kebutuhan sistem

| Actor   | Kebutuhan Sistem   |
|---------|--|
| Petugas | • Pengelola dapat menampilkan data buku                    |
|         | • Pengelola dapat menampilkan data mahasiswa               |
|         | • Pengelola dapat menampilkan data fakultas                |
|         | • Pengelola dapat menampilkan data program studi           |
|         | • Pengelola dapat menampilkan data kategori buku           |
|         | • Pengelola dapat menampilkan dan memproses data warehouse |
|         | • Pengelola dapat menganalisa data.                        |
|         | • Pengelola dapat menampilkan informasi tentang sistem.    |
|         | • Pengelola dapat keluar dari sistem.                      |

## 2.2. Perancangan Sistem

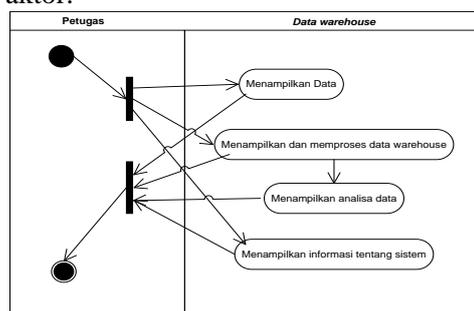
Actor yang berperan dalam sistem ini adalah petugas. Petugas merupakan staf dari perpustakaan, dalam hal ini merupakan staf yang diberi kewenangan dalam pengelolaan data warehouse. Perancangan diawali dengan pengembangan use case diagram. Use case diagram merupakan diagram yang memperlihatkan hubungan antara aktor dengan sistem (Whitten, 2004). Perancangan use case diagram untuk pengembangan sistem dapat dilihat pada Gambar 1. Dari gambar 2

memperlihatkan aktor yang berinteraksi dengan sistem, yaitu petugas. Aktor petugas mempunyai hak untuk menampilkan berbagai macam data, seperti data buku, data mahasiswa, data fakultas, data program studi dan data kategori buku. Aktor petugas juga dapat menampilkan dan memproses data warehouse. Selain itu, aktor petugas juga dapat menampilkan analisa data, menampilkan informasi tentang sistem dan dapat keluar dari sistem.



Gambar 1 Use case diagram

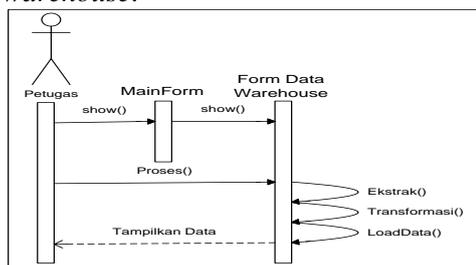
Selanjutnya dikembangkan activity diagram seperti dapat dilihat pada gambar 2. Activity diagram merupakan salah satu cara untuk memodelkan event-event yang terjadi dalam suatu use case (Whitten, 2004). Pada tahap ini dibuat perancangan activity diagram untuk masing-masing aktor.



Gambar 2 Activity diagram

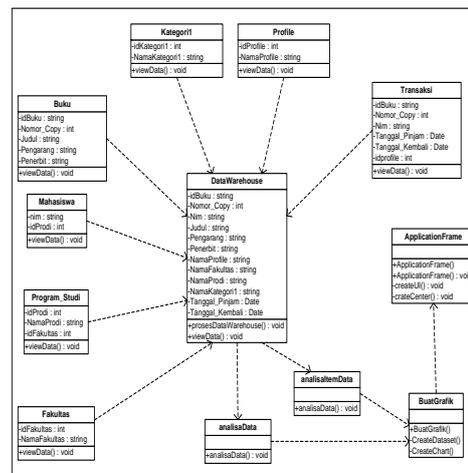
Diagram selanjutnya yang dikembangkan adalah sequence diagram. Sequence diagram merupakan interaction diagram yang memperlihatkan event-event yang

berurutan sepanjang berjalannya waktu (Whitten, 2004). *Sequence* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah *event* untuk menghasilkan output tertentu. Pada gambar 3 dapat dilihat *sequence diagram* untuk memproses *data warehouse*. Setelah FormUtama ditampilkan, petugas dapat menampilkan form *data warehouse*. Pada form *data warehouse*, akan dilakukan proses ekstraksi data dari data sumber. Proses ekstraksi ini adalah memilih *field-field* tertentu dari data sumber. *Field-field* ini kemudian akan digabungkan melalui proses transformasi untuk selanjutnya akan disimpan ke dalam sebuah file eksternal. File eksternal merupakan *load file* yang nantinya akan disimpan ke dalam *data warehouse*.



Gambar 3.5 *Sequence diagram data warehouse*

Diagram terakhir yang dikembangkan adalah *class diagram*. *Class diagram* adalah diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem/ perangkat lunak yang sedang dikembangkan (Whitten, 2004). Pada gambar 4 ditunjukkan *class diagram* untuk sistem yang akan dikembangkan.



Gambar 4 *Class Diagram*

### 2.3. Implementasi

Tahapan implementasi merupakan tahapan pengembangan aplikasi. Pada model *prototyping*, tahapan implementasi dapat dipandang sebagai tahapan pengembangan *prototype*. *Prototype* merupakan gambaran utuh dari sistem yang akan dibangun. *Prototype* masih memungkinkan untuk dievaluasi untuk disempurnakan agar dapat menjawab kebutuhan. Jika mencermati *class diagram*, maka dapat dilihat bahwa *class DataWarehouse* merupakan rangkuman dari beberapa *class*, yaitu mahasiswa, buku, transaksi, program studi, fakultas, kategori dan profile. Berikut adalah tahapan yang dilakukan untuk implementasi *data warehouse*:

- 1) Pembuatan tabel *DataWarehouse*

Skrip untuk proses pembuatan tabel dapat dilihat pada gambar 5.

```

1 DROP TABLE IF EXISTS `simpus`.`datawarehouse`;
2 CREATE TABLE `simpus`.`datawarehouse` (
3   `idbuku` varchar(200) NOT NULL,
4   `judul` text NOT NULL,
5   `pengarang` varchar(200) NOT NULL,
6   `penerbit` varchar(200) NOT NULL,
7   `nomor_copy` int(11) NOT NULL,
8   `namafakultas` varchar(100) NOT NULL,
9   `namaprodi` varchar(100) NOT NULL,
10  `namaprofile` varchar(10) NOT NULL,
11  `namakategori` varchar(50) NOT NULL,
12  `tanggal_pinjam` date NOT NULL,
13  `tanggal_kembali` date NOT NULL,
14  `nim` varchar(9) NOT NULL
15 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
    
```

Gambar 5 Skrip pembuatan *data warehouse*

2) Ekstraksi data

Pada proses ekstraksi data ini, akan diekstrak data dari tabel-tabel sumber yaitu data mahasiswa, data buku, data kategori1, data program studi, data fakultas dan data profile. Hasil ekstraksi ini akan disimpan dalam simpanan sementara. Listing program untuk proses ekstraksi data dapat dilihat pada gambar 6.

Dari gambar 6 dapat dilihat bahwa data-data diekstrak melalui proses kueri. Kueri ini disimpan dalam sebuah view bernama DW. Kondisi yang menjadi syarat dari kueri ini sendiri adalah bahwa data transaksi yang akan diekstrak merupakan transaksi yang terjadi setelah tanggalAkhir. Variabel tanggalAkhir merujuk pada tanggal terakhir dari data transaksi yang disimpan pada data warehouse. Selanjutnya data hasil ekstraksi akan disimpan dalam sebuah file berekstensi txt, yaitu pindah.txt.

```

stdW.execute("DROP VIEW IF EXISTS DW;");
stdW.execute("create view DW as select trim(a.idbuku) as idbuku,"
+ "trim(a.judul) as judul,trim(a.pengarang) as pengarang,"
+ "trim(a.penerbit)as penerbit,a.nomor_copy,d.namaaprodi,"
+ "e.namafakultas,g.namaprofile,f.namakategori,b.tanggal_pinjam,"
+ "b.tanggal_kembali,c.nim "
+ "from buku a, transaksi b, mahasiswa c, program_studi d, "
+ "fakultas e, kategori f, profile g "
+ "where a.idbuku=b.idbuku and a.nomor_copy=b.nomor_copy and b.nim=c.nim "
+ "and c.idprodi=d.idprodi and d.idfakultas=e.idfakultas "
+ "and b.idkategori=f.idkategori "
+ "and b.idprofile=g.idprofile "
+ "and b.tanggal_pinjam>"+tanggalAkhir+" order by b.tanggal_pinjam;");

ResultSet rsCek = stdW.executeQuery("select * from dw");
rsCek.beforeFirst();
if (rsCek.next()){
    this.lInfo.setText("Ekstrak data selesai");
    this.lInfo.setText("Store data to txt file.....");
    stdW.execute("select * into outfile 'pindah.txt' "
+ "fields terminated by '$';" from dw;");
}
    
```

Gambar 6 Listing Program ekstraksi data

3) Transformasi

Pada proses transformasi ini dilakukan pembersihan data dari duplikasi dan kesalahan data. Proses ini sebenarnya menjadi satu dengan proses ekstraksi data, di mana pada saat dilakukan kueri, sekaligus dilakukan pembersihan duplikasi dan kesalahan data. Di mana data-data yang memenuhi kondisi yang disyaratkan pada kueri hanyalah

data-data yang valid dan sudah tidak terdapat lagi duplikasi data.

4) Loading/ pengisian data ke dalam data warehouse

Proses selanjutnya adalah mengisikan data hasil ekstraksi dan transformasi ke dalam data warehouse. Listing program untuk proses ini dapat dilihat pada gambar 7. Dari gambar 7 dapat dilihat bahwa data diambil dari file pindah.txt kemudian disimpan ke dalam tabel DataWarehouse. Baik proses penyimpanan hasil ekstraksi data ke simpanan sementara dalam file pindah.txt maupun proses pengisian data yang diambil dari file pindah.txt ke dalam data warehouse dilakukan dengan memanfaatkan fasilitas pada MySQL. Perintah 'load data infile' sendiri dimaksudkan untuk proses penyimpanan data ke dalam tabel dengan kecepatan yang sangat tinggi. Hal ini diperlukan mengingat bahwa data yang diproses merupakan data dalam jumlah yang sangat besar, untuk itu diperlukan kecepatan pemrosesan. Kecepatan pemrosesan akan sangat mempengaruhi performa sistem.

```

stdW.execute("load data infile 'pindah.txt' "
+ "into table datawarehouse fields terminated by '$';");
    
```

Gambar 7 Listing program untuk pengisian data

3. HASIL DAN PEMBAHASAN

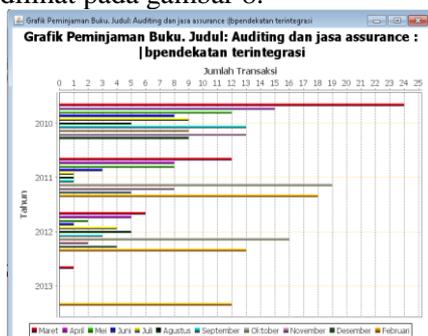
Dengan menggunakan data warehouse, maka akan dapat dilakukan berbagai analisa yang kompleks yang tidak memungkinkan untuk dilakukan dengan menggunakan CBIS. Aplikasi data warehouse biasanya dilengkapi dengan fitur untuk menampilkan informasi dalam bentuk grafik, sehingga memudahkan pengguna untuk membaca dan memahami informasi yang disajikan oleh sistem.

Analisa-analisa yang dilakukan tentu saja menyesuaikan dengan

kebutuhan organisasi. Analisa-analisa ini dapat dilakukan menggunakan kueri yang kompleks, juga dapat dilakukan dengan menggunakan kaidah-kaidah tertentu. Secara umum, dengan adanya gudang data atau *data warehouse*, maka memungkinkan untuk dikembangkan menjadi sebuah aplikasi *data mining* yang memang ditujukan untuk penggalian berbagai informasi tersembunyi dari data yang tersimpan dalam *data warehouse*.

Contoh analisa yang umum dibutuhkan oleh perpustakaan antara lain:

- 1) Analisa per judul buku  
Analisa ini dipergunakan untuk menganalisa jumlah transaksi peminjaman sebuah buku per tanggal pada setiap tahun transaksi peminjaman. Contoh grafik yang dihasilkan untuk analisa ini dapat dilihat pada gambar 8.



Gambar 8 Grafik analisa per judul buku

- 2) Analisa peminjaman oleh mahasiswa dari program studi tertentu terhadap kategori buku tertentu  
Analisa ini dipergunakan untuk menampilkan tren jumlah transaksi peminjaman buku oleh mahasiswa suatu program studi terhadap kategori buku tertentu. Contoh grafik untuk analisa ini dapat dilihat pada gambar 9.



Gambar 9 Grafik Analisa peminjaman oleh mahasiswa dari program studi tertentu terhadap kategori buku tertentu

#### 4. KESIMPULAN

Penyimpanan data transaksi pada perpustakaan dalam jangka waktu yang lama akan menyebabkan penumpukan data yang sudah tidak terpakai untuk transaksi dalam jumlah yang sangat banyak. Dengan membangun sebuah *data warehouse* atau gudang data, penumpukan data ini akan dapat mendatangkan nilai manfaat, di mana dapat dilakukan proses penggalian informasi dari data-data tersebut.

*Database Management System* (DBMS) biasanya sudah dilengkapi dengan fitur-fitur yang dapat berguna dalam membangun sebuah *data warehouse*. Fitur-fitur ini dapat bermanfaat dalam proses ekstraksi, transformasi serta loading atau pengisian data. Proses *information delivery* dapat disajikan dengan metode *complex queries*. *Complex queries* merupakan kueri atau pemilihan data tertentu menggunakan statemen SQL (*Structured Query Language*) pada DBMS di mana kueri tersebut bersifat kompleks, melibatkan beberapa tabel dan memanfaatkan fungsi agregasi dalam statemen SQL.

#### 5. SARAN

Agar sistem dapat bangun betul-betul bisa menjawab kebutuhan dan bisa merepresentasikan kondisi terkini serta analisa dalam proses *information*

*delivery* bisa menghasilkan informasi yang valid dan *up to date*, maka sistem sebaiknya diintegrasikan dengan sistem operasional perpustakaan. Selain itu, untuk mendapatkan informasi yang lebih bervariasi serta dapat menjawab berbagai kebutuhan, *data warehouse* dapat dipergunakan untuk mengembangkan aplikasi *data mining*.

#### DAFTAR PUSTAKA

- Poe, V., 1998, *Building Data Warehouse for Decision Support*, Prentice Hall, New Jersey.
- Pressman, Roger S., 2001, *System Engineering: A Practitioner's Approach, Fifth Edition*, McGraw Hill, New York
- Whitten, Jeffery L., Bentley, Lonnie D., Dittman dan Kevin C., 2004, *Systems Analysis Design Methods, sixth edition*. McGraw Hill, New York.